Dear reviewers,

We sincerely thank you for your valuable comments and suggestions, which have significantly contributed to enhancing the quality of our revised paper (Tracking number: 948-1972) and of the revised REASONX implementation. Below, you will find our detailed responses in blue, along with the corresponding revisions made to the manuscript. The revised paper also includes in blue the new/revised contents.

With best regards,
Laura State, in behalf of all authors

Reviewer #1

Firstly, I will note that I like the idea of REASONX, tackling often overlooked XAI desiderata like interactivity and working with underspecified samples.

We thank the reviewer for this positive assessment.

That being said, REASONX itself seems to have been published in the earlier works by the authors, and thus is not novel. The remaining contributions of the paper are (1) the algebra of operators over sets of linear constraints and (2) empirical evaluations. I found the algebra technically sound, but poorly motivated and generally lacking a goal. It is unclear to me if it is useful in any way, though admittedly, it is outside my expertise. The qualitative evaluations showcase the usage of REASONX well, though they are very similar to those in previously published work.

We agree that the REASONX prototype and its basic architecture were introduced in our earlier works, which appeared in conference proceedings. The present paper is, however, a substantial extension of those works, further strengthened by the additions made in the revised manuscript in response to this and the other reviewers' comments. In particular, we highlight: (1) a substantially revised motivation and positioning of the approach, together with a broader review of related work; (2) the formal definition of the query algebra and its operators, including the formulation of explanation problems as queries in this algebra; (3) a considerably more detailed presentation of the architecture, substantial extensions of the implementation, including the newly introduced fidelity checks (see later answer); and (4) a more systematic evaluation and comparison with other XAI methods. The main text of the revised manuscript is now 41 pages (four pages more than in the initial submission) and contains approximately 50% new material compared with the earlier conference works.

Going point-by-point:
- claim of model agnosticism - REASONX is not model agnostic.

Thanks for pointing out this presentation ambiguity. We now stress throughout the paper (and in the introduction, in particular) that our approach is model specific to DTs, and it can be *conditionally* considered model-agnostic (locally or globally) under the assumption of high fidelity. Fidelity of surrogate DTs used in experiments is reported in our results. We also point out that higher fidelity can be achieved at the cost of more complex explanations by training deeper DTs.

Training a surrogate model and explaining it (without even evaluating feasibility in the original model) is a trivial solution to the problem of having a model-specific explanation tool. This can be easily done with any other model-specific tool.

Only explainable models can be used as surrogates of black boxes. This is a common strategy followed by other post-hoc, “model-agnostic” methods such as LORE (using a decision tree, https://doi.org/10.1007/s10618-022-00878-5) or LIME (using a linear model, https://arxiv.org/abs/1602.04938). For local explanations, this generally works well, as in the neighbourhood of a specific instance, simple (explainable) models show high fidelity, as now highlighted in the supplemental material experimental section “More on Quantitative

Evaluation" (see Fig. 7 right). We now mention this in the beginning of the Related Work, in addition to the existing mention in the "REASONX architecture" subsection. Regarding evaluating feasibility, please see below.

I could not find even a discussion of how the tree surrogates were learned (code in the appendix suggests scikit-learn, i.e., CART), let alone suggestions or studies of different approaches and their consequences.

The subsection "From Python to CLP: Embeddings" included the characterization of the DTs that are admissible by the query language. Axis parallel and equality splits used in CART and C4.5, linear splits used in oblique and optimal trees, and axis parallel splits at nodes and linear splits at leaves used in linear-tree models are all covered. We have now added that CART and linear-tree models are dealt with by REASONX, and that experiments fix CART as base models.

Not evaluating the true fidelity of the generated CEs is unacceptable when the claim is that the approach is model agnostic. The reported feasibility (assumed to be computed on test set data) is not informative, since REASONX generates CEs out of distribution and close to the decision boundary (as exemplified by the qualitative examples). The actual counterfactual feasibility might thus differ a lot.

We agree this was the main drawback of the paper. We have implemented a fidelity check of contrastive explanations in the case of DT that are surrogate models of black-boxes. The theory part assumes some estimation method. The implementation part discusses our implementation that relies on MCMC sampling from polytopes, a problem with several algorithms in the literature and software library implementations. For details, please see the subsection "Estimating fidelity of contrastive explanations" in the "Python Layer" section.

- weak comparison - Only two established, but old methods DiCE and ANCHORS, even when newer ones are discussed in Related works. Still, from the older methods, Russell, C. (2019) might be more appropriate, as it handles diversity using MILP and might thus lead to similar CEs.

We acknowledge that we selected two older methods for a comparison with REASONX. We chose them for the following reasons: 1) they are well known, including their limitations, and can be straightforwardly integrated into Python scripts; 2) each method produces a different type of explanation that are both part of REASONX: while DiCE creates contrastive explanations and allows for the integration of constraints (including ones on the diversity of the contrastive instances), ANCHORS produces rules as explanations. We acknowledge that a comparison with the method by Russell (2019, https://arxiv.org/pdf/1901.04909v1) could have been a useful addition. The method was already included in Table 1. In the revised subsection "Contrastive Explanations" of the Related Work, we now discuss in more detail the differences with REASONX. In the revised manuscript, we clarify the scope of the evaluation, and moderate the presentation of the experiments as comparisons of selected capabilities rather than as a general state-of-the-art benchmark. See beginning of Section "Quantitative Evaluation".

- lack of clarity - The algebra of operators over sets of linear constraints has unclear motivation and benefits

We thank the reviewer for noting that we had taken the motivation and benefits of the algebra for granted. The purpose of the algebra is to provide a uniform, declarative, and compositional language for explanation problems: decision-tree paths, instances, background knowledge, and relationships among instances or models are represented as theories of linear constraints and manipulated through the same operators. Declarativeness separates the logical specification of a query from its physical evaluation. Closure allows query results to be composed and reused. Moreover, the CLP meta-interpreter provides a declarative implementation whose clauses closely follow the formal semantics, facilitating a correctness argument; a procedural implementation would instead require a separate proof of correspondence. We revised the Introduction, the initial part of section “A Query Language over Linear Constraints for XAI” and the initial part of subsection “CLP Layer and the Meta-interpreter” to clarify these aspects.

- and the empirical evaluation is presented in a confusing way. For example, columns in Table 8 share values of metrics that are (by authors admission) not equivalent, making the comparison difficult. Many common questions are left unanswered. On what data was fidelity computed? How many samples were used to train the models? How many factuals were used to generate CEs? How were the surrogate models trained?

We have substantially reworked the quantitative experiments, both in the main text and in the supplemental material. The supplemental material contains a more linear presentation of experimental settings, including datasets, metrics and parameters, settings for the demonstrators, and settings for the quantitative experiments, as well as the replacement of old additional experiments with a new section (“More on quantitative evaluation”) with scalability results (see next answer). The main text on “Quantitative Evaluation” has been also heavily revised in text and in results following: (1) speedup up to 50x in runtime due to the adoption of the Janus library, a bi-directional bridge that provides high-throughput inter-language calls between Prolog and Python (Swift and Andersen, 2023); (2) implementation of the calculation of factual rule coverage, as to allow for comparing with ANCHORS.

- omitted scalability discussion - trees of what size can REASONX handle? In discussion is the highest depth 5. This is a problem since at that depth, the tree is understandable by itself and leads to at most 32 rules. How many rules can the system handle?

The revised section “Additional Experiments” (previously “Parameter Testing”) in the Supplemental material now reports several analyses at the variation of the maximum tree depth, including elapsed time (scalability), number of leaves, accuracy/fidelity, etc. Overall, these experiments indicate that smaller base DTs should be preferred whenever possible. This is particularly favorable for local surrogate models, for which our experiments show that high fidelity can already be achieved with comparatively shallow and small trees. For global surrogate models, instead, fidelity must be traded off against increasing explanation size and computational cost.

Minor issues and questions:

- The algebra of operators is sound and an interesting perspective, but is it useful in any way? Wouldn't simply considering polyhedra work just as well?

See previous answer on the clarity of the algebra.

- I found the use of the term "linear constraints" for actual linear constraints and their conjunction (a polyhedron) quite unintuitive.

When a single (in)equality is specifically required, we use “primitive linear constraints”. When a conjunction of zero or more primitive linear constraints is required, we use “linear constraints”. The latter covers the former as a special case. And, more importantly, we almost never specifically require primitive linear constraints: REASONX accepts conjunctions of inequalities in any place a constraint can be passed.

- Given the method's goal of interactivity, why was a user study not performed?

Thank you for your comment. A user study was beyond the scope of our work. We acknowledge in the section “Limitations and Future Work”.

- Can REASONX consider rule sets, or do the rules need to be "non-overlapping," leading to the choice of trees?

Since the embedding of decision trees is actually a rule set, rule set models can themselves be potentially considered. However, in case of overlapping rules, some rule-voting mechanism is typically adopted to make predictions. Such a mechanism is not accounted for in the algebra as it would compromise compositionality, hence we cannot apply it to overlapping rule sets. We now mention this in the subsection “From Python to CLP: Embeddings”.

- Would your approach fall into the "formal XAI" (Marques-Silva, 2022) when looking at it as model specific method for decision trees?

The approach does not belong to formal XAI even for model-specific methods. Differences with formal XAI were discussed in the Related Work and in the supplemental material, section “Contrasting to Sufficient Reasons”.

- I understand the idea about "highest predictive uncertainty being at the decision splits", but in discrete values (common in tabular data), such a small relaxation can have a major effect. E.g., relaxing a rule x > 0 for a binary x would lead to no split at all. In your case, the ordinal values seem to be encoded in a way that allows for this. Is that not an issue?

First, we have revised the definition of the *relax* operator in the general form with the $\epsilon > 0$ value already accounted for. Such a value should be small, e.g., machine precision, but non-zero so as not to contradict the DT classification. Thus the statement about uncertainty has been removed, see subsection “On the relax operator”. Specifically for discrete values, there is no issue, since they are encoded as integers. Split conditions in decision trees typically take the form e.g., Gender_Male > 0.5, where 0.5 is the midpoint between 0 and 1. Such conditions are relaxed to Gender_Male >= $0.5+\epsilon$. Even if the split condition would be

Gender_Male > 0, the relaxation Gender_Male >= ϵ would not include incorrect values as far as ϵ > 0.

- This approach has one further limitation not mentioned (in an otherwise thorough discussion of limitations), namely model privacy. REASONX enables querying the decision boundary directly (by looking at the returned results), including listing the counterfactual leaves. This would be a major hurdle to adoption in client-facing applications, where it could lead to leaking the decision model entirely.

Thank you for your comment. Model and also data privacy are a general concern to XAI methods. We acknowledge this with an additional statement in the limitation section of our paper.

- In the example at the end, in the first reply by REASONX, how can the models assign the same label, but the rules have no intersection? The instance itself is proof of the non-emptiness of the intersection.

Thank you for spotting this error. We have corrected it by adapting the first answer of REASONX in the dialogue accordingly.

- Finally, the usefulness of the counterfactuals found by this method might be hindered by their being in a low-density region (e.g., as shown in Figure 5). Could REASONX include constraints on the (counter)factual plausibility?

Yes, any constraint can be used in REASONX, as long as it is linear. For example, upper and lower bounds could be used to enforce plausibility of the contrastive instance. Such constraints agree with what Karimi et al. 2022 (https://dl.acm.org/doi/10.1145/3527848) write in their paper when explaining plausibility constraints via domain consistency: “domain-consistency restricts the counterfactual instance to the range of admissible values for the domain of features”.

- Confusing dot notation, main j-th feature of i-th instance I_i.x_j, instead of the standard x_ij

We prefer the used notation instead of the matricial one, as it allows to denote a whole instance independently of its features, e.g., as I_i in the inst() operator.

Reviewer #2

Detail Comments
1. Despite the authors' disclaimer that the fidelity of surrogate DTs to the original black-box model must be assumed (page 7), the implications of imperfect fidelity require deeper investigation:
- Some (e.g., contrastive) explanations can be invalidated by the discrepancy. The risk is exacerbated by REASONX's capability to reason over under-specified instances - such instances cover broader geometric regions, which are more likely to intersect with areas of low surrogate fidelity.

We agree this was the main drawback of the paper. We have implemented a fidelity check of contrastive explanations in the case of DT that are surrogate models of black-boxes. The theory part assumes some estimation method. The implementation part discusses our implementation that relies on MCMC sampling from polytopes, a problem with several algorithms in the literature and software library implementations. For details, please see the subsection ”Estimating fidelity of contrastive explanations” in the “Python Layer” description.

- The framework could partially address the fidelity issue by, e.g., identifying and filtering out DT's decision paths known to have lower fidelity to the black box (and not using them in the explanation).
Could the authors briefly elaborate on the points mentioned above?

The user was already able to filter paths based on confidence of the prediction, which in the case of surrogate models means fidelity of the prediction. Through the implemented fidelity check, the estimated fidelity can be reasoned programmatically to filter out contrastive rules with estimated fidelity smaller than a given threshold.

2. While REASONX is technically an agnostic method under the authors' assumptions (page 4, 18), the restriction to (surrogate) DTs is highly restrictive and not fully supported by neurosymbolic literature (some works, e.g., extract logical rules and programs directly from black boxes). Are there other inherently interpretable models that could be compatible with the framework?

Since the embedding of decision trees is actually a rule set, rule set models can themselves be potentially considered. However, in case of overlapping rules, some rule-voting mechanism is typically adopted to make predictions. Such a mechanism is not accounted for in the algebra as it would compromise compositionality, hence we cannot apply it to overlapping rule sets. We now mention this in the subsection “From Python to CLP: Embeddings”.

3. Since the authors report a generally longer runtime of REASONX compared to other XAI methods (page 34), it would be useful to present basic asymptotic complexity analysis. How does REASONX fare against other methods, disregarding the implementation efficiency?

In the revised manuscript, we clarify the scope of the evaluation, and moderate the presentation of the experiments as comparisons of selected capabilities rather than as a general state-of-the-art benchmark. See beginning of Section “Quantitative Evaluation”. The main strengths of REASONX lie in its declarative formulation, high-level abstraction

mechanisms, and support for interactivity, rather than in computational efficiency. As an analogy, programming languages are often designed to favor different objectives: some prioritize execution efficiency, whereas others emphasize abstraction and expressiveness.

Regarding asymptotic complexity, the most computationally demanding operator is projection. Using Fourier--Motzkin elimination, its worst-case complexity is O(m^{2.5n}), where m is the number of linear inequalities and n the number of variables; see Jing et al. (2020, https://doi.org/10.1007/978-3-030-60026-6_16). However, we believe that a detailed complexity analysis would substantially complicate the presentation (running times are already included among the experimental results) and would be somewhat orthogonal to the main contribution of the paper. We therefore prefer not to include such a discussion in the manuscript.

Finally, the revised implementation of REASONX achieves now a speedup up to 50x in runtime due to the adoption of the Janus library, a bi-directional bridge that provides high-throughput inter-language calls between Prolog and Python (Swift and Andersen, 2023). The previous version relied on slow process calls and shared files.

4. Some of the implementation details seem to hinder efficient interactive querying (meaning, without full recomputation; useful in interactive settings, especially in the case of longer runtime). One of them is probably the usage of the single-answer MILP method bb_inf (page 23). How complex would the changes be, allowing efficient iterative usage of REASONX?

We agree that the current implementation supports interactivity primarily at the query-language level, rather than through systematic reuse of solved (sub-)expressions across successive queries. We mention this in the subsection “Interactivity” of the Related Work section, also in contrast to a newly cited incremental method (UGCE, https://arxiv.org/abs/2505.21330v1) for contrastive explanations. The effort required to improve iterative use depends on the desired level of incrementality. Caching previously computed subexpressions results would be comparatively straightforward and could already reduce recomputation for repeated or structurally similar queries. In contrast, true incremental MILP optimization, in which branch-and-bound information or solver state is retained across successive queries, would require a more substantial change, since this functionality is not exposed by the current bb_inf interface of CLP(R). It would likely require either extending the underlying solver interface or integrating a MILP backend that supports persistent models, warm starts, and incremental addition or removal of constraints. Such extensions could be addressed as future work.

5. The following claim should be softened: REASONX is working with "any black-box predictor" (abstract) - DTs are inappropriate surrogates for some models, especially for models working with non-tabular data.

Thank you for your comment. We have adjusted the abstract (and the paper) to restrict to tabular-data classifiers. Moreover, we now stress throughout the paper that our approach is model specific to DTs, and it can be *conditionally* considered model-agnostic (locally or globally) under the assumption of high fidelity. Fidelity of surrogate DTs used in experiments is reported in our results.

Reviewer #3

The paper spends, in my opinion, too many pages describing implementational details of comparatively little interest (the discussion of the meta-interpreter at pages 20-23 being perhaps an especially egregious example); instead, the "evaluation" part of pages 31-onwards, which should be the most important one since the paper is about presenting a new tool, contains only a limited discussion of the results and leaves many important details in the appendix, to the point of being borderline unparseable as it is.

We have substantially reworked the quantitative experiments, both in the main text and in the supplemental material. The supplemental material contains a more linear presentation of experimental settings, including datasets, metrics and parameters, settings for the demonstrators, and settings for the quantitative experiments, as well as the replacement of old additional experiments with a new section (“More on quantitative evaluation”) with scalability results (see next answer). The main text on “Quantitative Evaluation” has been also heavily revised in text and in results following: (1) speedup up to 50x in runtime due to the adoption of the Janus library, a bi-directional bridge that provides high-throughput inter-language calls between Prolog and Python (Swift and Andersen, 2023); (2) implementation of the calculation of factual rule coverage, as to allow for comparing with ANCHORS.

For space reasons, however, we leave experimental details and experiments on the parameters of REASONX in the supplemental material. In the revised manuscript, we clarify the scope of the evaluation, and moderate the presentation of the experiments as comparisons of selected capabilities rather than as a general state-of-the-art benchmark. See beginning of Section “Quantitative Evaluation”. The main strengths of REASONX lie in its declarative formulation, high-level abstraction mechanisms, and support for interactivity, rather than in computational efficiency. At the same time, we prefer to retain the discussion of the meta-interpreter in the main text. In the context of a journal on neurosymbolic AI, we consider it an important contribution rather than a purely implementational detail. The meta-interpreter provides the executable realization of the algebra and makes explicit the correspondence between the declarative semantics of the operators and their implementation in CLP.

* Page 4: "---whether the explanation method exploits knowledge about the internals of the black-box or not". If we can see "the internal of the black box", isn't that *not* a black-box by definition?

A “black-box” is a model that is not interpretable by humans. This means such a model is not understandable to a human, for example, because it contains non-linearities. Nevertheless, one can have access to the internals of the model, e.g., to the weights of a neural network. We clarified this definition in the paper.

* Page 6: "Bayes rule". What is meant by this? Bayes' rule isn't about choosing the label with the highest estimated probability, as the phrase would seem to suggest; and it doesn't seem to be relevant to the framework described here, in which the leaves simply assign probabilities to labels.

Yes, Bayes' rule consists of choosing the class value with the highest estimated probability. This is the best possible choice given the estimated probability distribution. It is relevant to our framework, because we assume that leaves predict a class value, e.g., in the inst(I, DT, l, pr)

operator the user selects instance I from a leaf of DT with predicted class value l with confidence at least pr.

* Pages 11 and following: a criticism of the authors' notion of "factual explanation" that should be worth addressing, I think, is that depending on the number of variables and on the structure of the decision tree the explanation generated by the algorithm might involve a number of variables that are not *actually* relevant to the decision made and might make it difficult to parse for a human user. One can answer this in a few different ways (for example, by arguing that if so then the problem lies in the fact that the decision tree is badly constructed, or suggesting that this factual explanation might be simplified later on if necessary but at any rate it still need to be constructed first); but in any case, I believe that this should be addressed.

We agree that a factual rule obtained from a decision-tree path may contain features that are not strictly necessary to preserve the prediction, especially for deep or complex trees. REASONX reports the conditions actually encountered along the decision path, rather than computing a subset-minimal explanation. Redundant inequalities within the path are removed by the constraint solver (see next question), but a condition may still be unnecessary when alternative paths leading to the same prediction are taken into account. A natural alternative is given by *sufficient reasons*, which seek a minimal subset of feature values guaranteeing the same prediction. We already contrast the two notions in the Supplemental Material: sufficient reasons can yield more succinct explanations, but multiple minimal explanations may exist and their computation is generally intractable; moreover, their standard definition assumes fully specified instances, whereas REASONX is explicitly designed to reason also over under-specified ones. Finally, minimality in sufficient reasons would compromise compositionality of the algebra (this observation has been now added to the paper).

* Also pages 11 and following: something should also be said about how to address the fact that a set of constraints can often be made non-redundant in different ways. Presumably, we want to give the user the constraints in the most readable form possible: but, to make a fairly trivial example, how are we to choose between {x1 >=3, x2 >=2, x1 + x2 <= 5} and {x1 = 3, x2 = 2}? Intuitively in this case it is clear that the second set of constraints is preferable, but in general one needs a well-defined notion of what this means and a description of what the system will return.

Redundant inequalities are removed and constraints are simplified by the CLP solver automatically (“for free”). This is a well-studied problem in the constraint-solving literature, and we mentioned it in Section “Constraint Logic Programming over the Reals” of the Supplemental material - see references to Greenberg (1996), Bagnara et al. (2005), and Maréchal and Périn (2017). In the revised manuscript, we now also make this explicit at the beginning of subsection “CLP Layer and the Meta-interpreter.” Regarding your example, when the CLP(R) solver is given the input {x1 >=3, x2 >=2, x1 + x2 <= 5}, it returns {x1 = 3, x2 = 2}.

* Page 17: "the factual rules are produced only for c's that appear as c and c' \in Th(S) for some c'". It is not clear to me what is meant here. What is S? Does it refer to the previous point?

Thank you for this comment, which revealed that the original sentence was poorly phrased. Here, S denotes the query expression introduced in the previous point, whose evaluation

produces the answer constraints. During the evaluation of S, a sub-expression F_i=inst(I_i,DT_i,I_i,pr_i) generates a path constraint c, which forms the left-hand side of the corresponding factual rule. We have revised the text to make this relationship between S, the path constraint c, and the generated factual rule explicit.

* Page 19: "Each declared instance is encoded by a list of CLP(R) variables, one for each feature". Shouldn't it be encoded by a list of variable *values*?

The encoding indeed uses a list of CLP(R) variables, one for each feature. Their bindings to concrete values, when available, are introduced separately through the user constraints Φ. While this distinction was explicit in the section defining the algebra, it was less clear in the discussion of the embeddings. In the revised manuscript, we now state this explicitly when introducing the user constraints Φ.

* Pages 21-23: as I was saying, I think that the discussion of these implementational details could have been safely moved to the supplementary material.

See answer to the first comment.

* Page 32: I think that more discussion of Table 7 is necessary. Here we do not even see an explanation of what its various columns mean, never mind some insight about what these results tell us about the performance of the proposed tool. Similarly for the other tables in this section (I see that there is some material along these lines in the supplementary material, but it really should be part of the main paper).

See the answer to your first question. Captions of tables are now informative of the meaning of the columns. Moreover, the text discussing results provides insights about them.

* Page 35: "REASONX advances other methods in qualitative terms." Such as?

For brevity, we added a reference to Table 1. A more detailed account would highlight that REASONX provides a unified declarative framework for reasoning over factual and contrastive rules and contrastive examples, while supporting user-defined constraints, under-specified instances, multiple instances and models, and interactive refinement of explanation queries. However, this has been stated repeatedly throughout the paper.

* Page 48: Much of the "Contrasting to sufficient reasons" section should be moved in the main text, I think; and the same applies also to the "Quantitative Evaluation" section later on.

We agree that both sections contain material that is relevant to the main contributions of the paper. However, we prefer to retain them in the Supplemental material in order to preserve the flow and readability of the main text, which already introduces a substantial amount of formal and implementation details. In the revised manuscript, we have instead strengthened the connections to these sections by adding more explicit references in the main text, so that readers are directed to the discussion of sufficient reasons and to the quantitative evaluation at the points where they are most relevant.